## CLAIMS

What is claimed is:

1. A method for generating a digital or visual representation of a closed tessellated surface geometry comprising the steps of:

a. importing a geometric model into a model processor, the model having a geometry defining a shape of the model;

b. generating a volume mesh around the imported geometric model;

c. extracting a first mesh front that encloses the model, the first mesh front comprising a closed mesh that generally conforms to geometry of the model;

d. mapping the first mesh front onto the model geometry; and

e. optimizing the first mesh front, thereby creating a digital representation of a closed tessellated surface geometry.

2. The method of claim 1 further comprising the step of exporting the model from the model processor.

3 The method of claim 2 where the model is exported for a visual representation on a display screen.

4. The method of claim 1 wherein the geometric model comprises a 2-D model.

5. The method of claim 4 wherein the volume mesh comprises a plurality of 2-D cells.

6. The method of claim 5 wherein the step of generating a volume mesh comprises generating a 2-D bounding box around the geometry and then filling the bounding box with a volume mesh.

7. The method of claim 5 wherein the volume mesh is a 2-D Cartesian mesh.

8. The method of claim 6 wherein the first mesh front comprises at least one collection of a plurality of faces.

9. The method of claim 5 wherein the step of extracting a first mesh front includes

identifying and discarding all cells that intersect the geometry of the model, thereby defining a first collection of cells positioned inside the geometry and a second collection of cells positioned outside the geometry,

discarding one of the first or the second collection of cells, thereby defining at least one hole in the volume mesh, the hole enclosing a part in the geometric model, and

identifying hole boundaries associated with the hole and grouping the hole boundaries into a collection of faces, the collection of faces defining a mesh front.

10. The method of claim 9 wherein the step of mapping the mesh front onto the model geometry comprises

eliminating sharp corners from the mesh front, and

smoothing the mesh front.

11. The method of claim 10 wherein the step of mapping the mesh front onto the model geometry further comprises projecting mesh vertices directly onto the geometry.

12. The method of claim 11 wherein the step of optimizing the mesh front includes smoothing the vertices and re-projecting the vertices onto the geometry.

13. The method of claim 10 wherein the step of optimizing the mesh front includes combining faces.

14. The method of claim 1 wherein the geometric model comprises a 3-D model.

15. The method of claim 14 wherein the volume mesh comprises a plurality of 3-D cells.

16. The method of claim 14 wherein the step of generating a volume mesh comprises generating a 3-D bounding box around the geometry and then filling the bounding box with a volume mesh.

17. The method of claim 16 wherein the volume mesh is a 3-D Cartesian mesh.

18. The method of claim 14 wherein the first mesh front comprises at least one collection of a plurality of faces.

19. The method of claim 15 wherein the step of extracting a first mesh front includes

identifying and discarding all cells that intersect the geometry of the model, thereby defining a first collection of cells positioned inside the geometry and a second collection of cells positioned outside the geometry,

discarding one of the first or the second collection of cells, thereby defining at least one hole in the volume mesh, the hole enclosing a part in the geometric model, and

identifying hole boundaries associated with the hole and grouping the hole boundaries into a collection of faces, the collection of faces defining a mesh front.

20. The method of claim 19 wherein the step of extracting a first mesh front further comprises identifying polygonal front faces and splitting the polygonal front faces into triangular or quadrilateral front faces.

21. The method of claim 19 wherein the step of mapping the mesh front onto the model geometry comprises

eliminating sharp corners from the mesh front, and

smoothing the mesh front.

22. The method of claim 21 wherein the step of mapping the mesh front onto the model geometry further comprises projecting mesh vertices directly onto the geometry.

23. The method of claim 22 wherein the step of optimizing the mesh front includes smoothing the vertices and re-projecting the vertices onto the geometry.

24. The method of claim 22 wherein the step of optimizing the mesh front includes combining faces.

25. The method of either claim 11 or claim 21 wherein the step of smoothing the mesh front comprises multiple smoothing passes.

26. The method of either claim 25 wherein the step of smoothing the mesh front further comprises using a Laplacian smoothing algorithm.

27. The method of claim 26 wherein each smoothing pass comprises projecting each mesh vertex onto the geometry of the model using a closest point projection.

28. Fluid dynamic simulation software for generating in a computer a visual or digital representation of a closed tessellated surface geometry comprising:

a. a first set of instructions functional to import a geometric model into a model processor operatively associated with the computer;

b. a second set of instructions functional to generate a volume mesh around the imported geometric model;

c. a third set of instructions functional to extract a first mesh front that encloses the model, the first mesh front comprising a closed mesh that generally conforms to geometry of the model;

d. a fourth set of instructions functional to map the first mesh front onto the model geometry; and

e. a fifth set of instructions functional to optimize the first mesh front.

29. The software of claim 28 further comprising a sixth set of instructions functional to export the model from the computer.

30. Computer software for generating in a computer a visual or digital representation of a closed tessellated surface geometry comprising:

a. computer code functional to import a geometric model into the computer, the model having a geometry defining a shape of the model;

b. computer code functional to generate a volume mesh around the imported geometric model;

c. computer code functional to extract a first mesh front that encloses the model, the first mesh front comprising a closed mesh that generally conforms to geometry of the model;

d. computer code functional to map the first mesh front onto the model geometry; and

e. computer code functional to optimize the first mesh front, thereby creating a digital representation of a closed tessellated surface geometry.

31. The software of claim 30 further comprising computer code functional to export the model from the computer.

32. The software of claim 31 where the model is exported for a visual representation on a display screen.

33. The software of claim 30 wherein the geometric model comprises a 2-D model.

34. The software of claim 33 wherein the volume mesh comprises a plurality of -2-D cells.

35. The software of claim 34 further comprising computer code functional to generate t he volume mesh by generating a 2-D bounding box around the geometry and then fill the bounding box with a volume mesh.

36. The software of claim 34 wherein the volume mesh is a 2-D Cartesian mesh.

37. The software of claim 35 wherein the first mesh front comprises at least one collection of a plurality of faces.

38. The software of claim 34 wherein the computer code that is functional to extract a first mesh front is further functional to

identify and discard all cells that intersect the geometry of the model, thereby defining a first collection of cells positioned inside the geometry and a second collection of cells positioned outside the geometry,

discard one of the first or the second collection of cells, thereby defining at least one hole in the volume mesh, the hole enclosing a part in the geometric model, and

identify hole boundaries associated with the hole and grouping the hole boundaries into a collection of faces, the collection of faces defining a mesh front.

39. The software of claim 38 wherein the computer code that is functional to map the mesh front onto the model geometry is further functional to

eliminate sharp corners from the mesh front, and

smooth the mesh front.

40. The software of claim 39 wherein the computer code that is functional to map the mesh front onto the model geometry is further functional to project mesh vertices directly onto the geometry.

41. The software of claim 40 wherein the computer code that is functional to optimize the mesh front is further functional to smooth the vertices and re-project the vertices onto the geometry.

42. The software of claim 39 wherein computer code that is functional to optimize the mesh front is further functional to combine faces.

43. The software of claim 30 wherein the geometric model comprises a 3-D model.

44. The software of claim 43 wherein the volume mesh comprises a plurality of 3-D cells.

45. The software of claim 43 wherein computer code that is functional to generate a volume mesh is further functional to generate a 3-D bounding box around the geometry and then fill the bounding box with a volume mesh.

46. The software of claim 45 wherein the volume mesh is a 3-D Cartesian mesh.

47. The software of claim 30 wherein the first mesh front comprises at least one collection of a plurality of faces.

48. The software of claim 44 wherein the computer code that is functional to extract a first mesh front is further functional to

identify and discard all cells that intersect the geometry of the model, thereby defining a first collection of cells positioned inside the geometry and a second collection of cells positioned outside the geometry,

discard one of the first or the second collection of cells, thereby defining at least one hole in the volume mesh, the hole enclosing a part in the geometric model, and

identify hole boundaries associated with the hole and grouping the hole boundaries into a collection of faces, the collection of faces defining a mesh front.

49. The software of claim 48 wherein the computer code that is functional to extract a first mesh front is further functional to identify polygonal front faces and split the polygonal front faces into triangular or quadrilateral front faces.

50. The software of claim 48 wherein the computer code that is functional to map the mesh front onto the model geometry is further functional to

eliminate sharp corners from the mesh front, and

smooth the mesh front.

51. The software of claim 50 wherein the computer code that is functional to map the mesh front onto the model geometry is further functional to project mesh vertices directly onto the geometry.

52. The software of claim 51 wherein the computer code that is functional to optimize the mesh front is further functional to smooth the vertices and re-project the vertices onto the geometry.

53. The software of claim 51 wherein the computer code that is functional to optimize the mesh front is further functional to combine faces.

54. The software of either claim 41 or claim 51 wherein the computer code that is functional to smooth the mesh front is further functional to perform multiple smoothing passes.

55. The software of claim 54 wherein the computer code that is functional to smooth the mesh front uses a Laplacian smoothing algorithm.

56. The software of claim 55 wherein each smoothing pass comprises projecting each mesh vertex onto the geometry of the model using a closest point projection.